



Bài tập lớn xử lý dữ liệu Python

Class: Data analyst with Python - 2324_AIT2003_1

Student name: Nguyễn Huy Hoàng

Student id: 22022584.

Project: Analysis of the Facebook fanpage

Fanpage: <https://www.facebook.com/Theanh28>

Github: https://github.com/hoanghelloworld/Final_Project_Python

Link notion :<https://determined-parakeet-21a.notion.site/B-i-t-p-l-n-x-l-d-li-u-Python-1af8c187c3da41e7a838911bc65e0518?pvs=4>

Mục lục

Lời nói đầu

Phần 1.Thu thập dữ liệu

1.Chuẩn bị thu thập dữ liệu

2.Thu thập dữ liệu

Phần 2.Làm sạch và tiền xử lý dữ liệu

1.Xóa các cột chứa đa số giá trị null hoặc ta không dùng đến khi phân tích

2.Sắp xếp lại dữ liệu theo thứ tự thời gian

3.Xử lý dữ liệu user like ,comment

4.Xây dựng các mô hình dự đoán

Phần 3.Phân tích dữ liệu

1.Phân tích về lượng tương tác của các bài viết

2.Phân tích nội dung bài viết

3.Phân tích user comment, user like

Kết luận

Lời nói đầu

Trong thời đại công nghệ số hiện nay, mạng xã hội là một trong những kênh truyền thông quan trọng và phổ biến nhất. Mạng xã hội không chỉ giúp mọi người kết nối, chia sẻ và giải trí, mà nó còn là nơi cung cấp cho chúng ta vô số thông tin về các vấn đề xã hội, chính trị, kinh tế, văn hóa, giáo dục, thể thao, nghệ thuật và nhiều lĩnh vực khác. Một trong những mạng xã hội phổ biến nhất hiện nay là Facebook, với hơn 2,8 tỷ người dùng trên toàn thế giới.



Facebook cho phép người dùng tạo ra các trang cá nhân, nhóm hoặc fanpage để thể hiện bản thân, quảng bá sản phẩm, dịch vụ, thương hiệu, sự kiện, tổ chức, cộng đồng hoặc các mục đích khác. Mỗi fanpage trên Facebook đều có những nội dung, hình ảnh, video, bình luận, lượt thích, chia sẻ và tương tác khác nhau từ người dùng. Những dữ liệu này đều có giá trị lớn đối với những nhà nghiên cứu, nhà quản lý, nhà phát triển, nhà kinh doanh và nhiều đối tượng khác, bởi vì chúng phản ánh được xu hướng, nhu cầu, hành vi, quan điểm và cảm xúc của người dùng đối với các chủ đề liên quan.

Trong bài báo cáo này, tôi sẽ sử dụng ngôn ngữ lập trình Python để thu thập, làm sạch, xử lý và phân tích dữ liệu từ một fanpage trên Facebook có tên là Theanh28. Đây là một fanpage nổi tiếng về tin tức thời sự và xã hội cũng như các nội dung giải trí, với hơn 10 triệu người theo dõi. Tôi sẽ cố gắng trả lời các câu hỏi sau:

- Fanpage Theanh28 có những nội dung gì? Chúng được phân bố như thế nào theo thời gian, loại bài viết, số lượng bình luận, lượt thích và chia sẻ?
- Người dùng có những phản ứng và tương tác như thế nào với các bài viết trên trang? Có những chủ đề nào được quan tâm, bàn luận và chia sẻ nhiều nhất?
- Các bài viết trên fanpage Theanh28 có ảnh hưởng đến cộng đồng mạng như thế nào? Có những ý kiến, quan điểm và cảm xúc nào được thể hiện qua các bình luận, lượt thích và chia sẻ?
- Những người nào đang tham gia dùng và tương tác với mạng xã hội

Để trả lời những câu hỏi trên, tôi sẽ sử dụng các thư viện và công cụ của Python như requests, BeautifulSoup, pandas, numpy, matplotlib, seaborn, nltk, wordcloud, textblob, v.v. Tôi hy vọng bài tập lớn này sẽ mang lại những kiến thức và kỹ năng hữu ích cho chúng ta - những người làm công nghệ nói riêng cũng như cho những người làm kinh tế quan tâm đến phân tích dữ liệu từ mạng xã hội nói chung.

Phần 1.Thu thập dữ liệu

1.Chuẩn bị thu thập dữ liệu

Cài đặt thư viện cần thiết

Tôi sẽ sử dụng thư viện facebook-scraping, selenium và apify để thu thập dữ liệu từ Facebook và sẽ cài đặt các thư viện này bằng pip.

```
%pip install facebook_scraper  
%pip install -U selenium  
%pip install apify-client
```

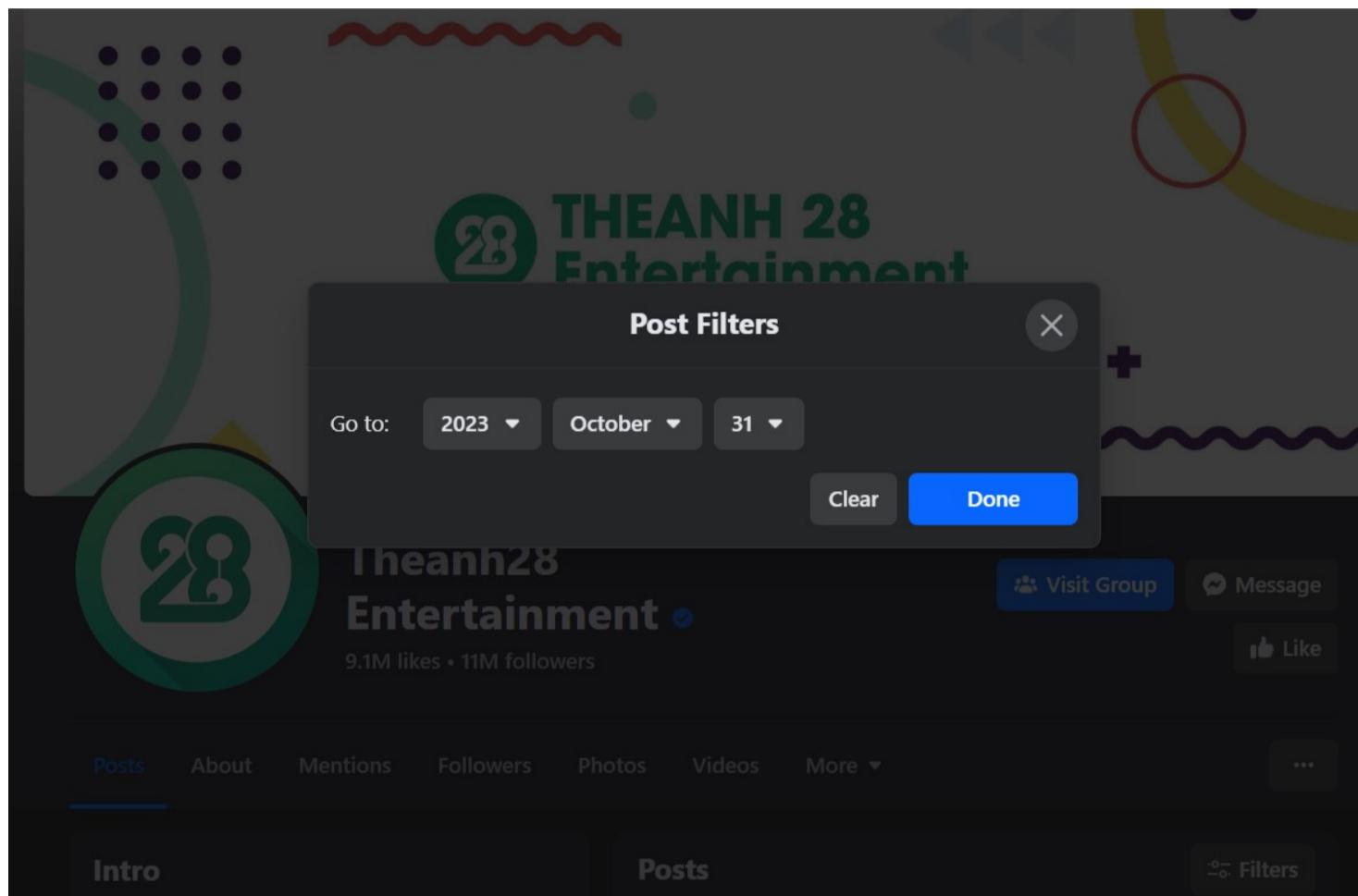
Cài đặt các thư viện làm việc với dữ liệu

```
%pip pandas numpy
```

2.Thu thập dữ liệu

2.1. Thu thập post_id

Vì đây là trang lớn nên có rất nhiều bài viết. sẽ lấy bài từ ngày 1/10/2023 đến ngày 31/10/2023, ước tính số lượng bài đăng dưới 2000. Đặt bộ lọc trên Facebook và trả đến ngày 31/10/2023 để thu thập dữ liệu từ đó(facebook_scraper chỉ hỗ trợ thu thập các bài viết gần nhất nên mình cần lấy dữ liệu từ ngày cuối)



```
driver = initDriver() # bắt browser  
fbLogin(driver, EMAIL, PASSWORD) # đăng nhập vào facebook  
sleep(25) # thời gian chờ để login  
getNumPost(driver, PAGE_ID, 2000) # lấy 2000 post id và lưu vào Data/data_post_id_1.csv
```

Phần code chi tiết tôi đã gắn trên github và sẽ đính kèm link theo bài báo cáo này

Thu thập các dữ liệu cơ bản của post(trừ user like,comment sẽ trình bày phần dưới)

Tôi sẽ sử dụng chức năng get_posts để lấy các bài đăng từ fanpage. Tìm thêm thông tin về thư viện facebook_scraper tại đây:

<https://github.com/kevinzg/facebook-scraper>

```
# Lấy dữ liệu  
options = {  
    "reactions": True,  
    "allow_extra_requests": True,  
    "reactors": False,  
    "progress": True,  
    "shares": False  
}  
post_=[]  
# lay du lieu facebook  
for post in get_posts(  
    post_urls= [f"https://facebook.com/{id}" for id in post_IDs],  
    options=options,  
    credentials=(EMAIL, PASSWORD)  
):  
    print(post)  
    post_.append(post)  
df=pd.DataFrame(post_)  
df.to_csv('Data/chuaxuly.csv', index=False) #chuyển dataframe thành định dạng csv
```

```
print(len(df))  
#output : 1651
```

Tôi thu được 1651 bài viết trong 1 tháng từ fanpage, một con số tương đối lớn

2.2. Thu thập về user like ,comment mỗi post

Giới thiệu thêm về APIFY:

Apify là một công cụ cho phép ta thu thập dữ liệu từ các trang web bất kỳ (facebook,tiktok,youtube,...),ta có thể cấu hình **Apify** để theo dõi các liên kết trang web và tạo ra một hàng đợi duyệt web để quy hoặc chỉ định một danh sách các URL cần duyệt. **Apify** sẽ tự động quản lý độ song song để đạt hiệu suất cao nhất.**Apify client** là một thư viện cho phép ta tương tác với nền tảng **Apify** bằng cách sử dụng các ngôn ngữ lập trình khác nhau, chẳng hạn như JavaScript hoặc Python. **Apify client** cho phép ta gọi các API của Apify để tạo, cập nhật, quản lý và giám sát các actor, task, dataset, request queue và các tài nguyên khác. Tìm hiểu thêm về **Apify** client tại [đây](#) hoặc [đây](#).

Lấy data comment user:

```
from apify_client import ApifyClient  
client = ApifyClient(token='apify_api_72Ye7UlGqQrkjq8GvyiQHzjMEhFFIt0zbKlh')
```

```
# Print len the dataframe  
print(len(df1))
```

```

run_input = {
    "startUrls" : [{"url": f"{url}"} for url in post_ids],
    "resultsLimit": 100
}

# Run the Actor and wait for it to finish
# .call method waits infinitely long using smart polling
# Get back the run API object
run = client.actor("apify/facebook-comments-scraper").call(run_input=run_input)

# Create an empty list to store the items
items_list = []

# Iterate over the items and append them to the list
for item in client.dataset(run["defaultDatasetId"]).iterate_items():
    items_list.append(item)

# Create a dataframe from the list
df1 = pd.DataFrame(items_list)

df1.to_csv ('Data/comments_pro.csv', index=False)

```

output:96454

Lấy data like user

```

# Initialize the ApifyClient with API token
client = ApifyClient("apify_api_72Ye7UlGqQrkjq8GvyiQHzjMEhFFIt0zbKlh")

# Prepare the Actor input
run_input = {
    "startUrls" : [{"url": f"{url}"} for url in post_ids],
    "resultsLimit": 19
}

# Run the Actor and wait for it to finish
run = client.actor("apify/facebook-likes-scraper").call(run_input=run_input)

# Create an empty list to store the items
items_list = []

# Iterate over the items and append them to the list
for item in client.dataset(run["defaultDatasetId"]).iterate_items():
    print(item)
    items_list.append(item)

# Create a dataframe from the list
df2 = pd.DataFrame(items_list)

df2.to_csv ('Data/likes_pro.csv', index=False)

```

```

# Print the dataframe
print(len(df2))
# output:25403

```

2.3 Thu thập dữ liệu user like,comment

```

list = []
for account in post_ids:
    try:
        profile = get_profile(account)
        list.append(profile)
        print(profile)
    except Exception as e:
        print(f"An error occurred with account {account}: {e}")
        continue
df = pd.DataFrame(list)
df.to_csv('Data/user.csv', header=False, index=False)

```

```

<class 'pandas.core.frame.DataFrame'>
RangeIndex: 18568 entries, 0 to 18567
Data columns (total 2 columns):
 #   Column           Non-Null Count  Dtype  
 ---  -- 
 0   id               18568 non-null   object 
 1   Nơi từng sống      10149 non-null   object 
 dtypes: object(2)
memory usage: 290.3+ KB

```

Phần 2. Làm sạch và tiền xử lý dữ liệu

Đọc dữ liệu chưa xử lý

```

df = pd.read_csv('Data/chuaxuly.csv')
df.info()

```

```

output :
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 1651 entries, 0 to 1650
Data columns (total 55 columns):
 #   Column           Non-Null Count  Dtype  
 ---  -- 
 0   original_request_url    1651 non-null   object 
 1   post_url                1651 non-null   object 
 2   post_id                 1651 non-null   object 
 ...
 53  videos                  269 non-null   object 
 54  header                  14 non-null    object 
 dtypes: bool(3), float64(14), int64(5), object(33)
memory usage: 675.7+ KB

```

1.Xóa các cột chứa đa số giá trị null hoặc ta không dùng đến khi phân tích

```

# Danh sách các cột ta xóa
columns_to_drop = ['comments_full', 'sharers', 'video_height', 'header', 'was_live', 'shared_text',
                    'video_duration_seconds', 'video_quality', 'video_size_MB', 'video_watches',
                    'video_width', 'is_live', 'factcheck', 'shared_post_id', 'shared_time', 'shared_user
                    _id',
                    'shared_username', 'shared_post_url', 'available', 'original_text', 'shared_text'
                    , 'timestamp', 'with', 'video_ids', 'videos', 'w3_fb_url', 'reactors', 'fetched_tim
                    e', 'image_id', 'link']

# Sử dụng phương thức drop
df = df.drop(columns=columns_to_drop)
df.info()

```

```

#output:df.info()
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 1651 entries, 0 to 1650
Data columns (total 26 columns):
 #   Column           Non-Null Count  Dtype  
 ---  -- 
 0   original_request_url    1651 non-null   object 
 1   post_url                1651 non-null   object 
 2   post_id                 1651 non-null   object 
 ...
 24  page_id                 1651 non-null   int64 
 25  image_ids                1651 non-null   object 
 dtypes: float64(2), int64(5), object(19)
memory usage: 335.5+ KB

```

2.Sắp xếp lại dữ liệu theo thứ tự thời gian

2.1. Lọc các dữ liệu chỉ lấy các bài viết trong tháng 10

```
# Chuyển đổi cột 'time' thành định dạng datetime
df['time'] = pd.to_datetime(df['time'])

# Lọc DataFrame để chỉ giữ lại các dòng trong tháng 10
df = df[df['time'].dt.month == 10]
```

2.2 Sắp xếp lại dữ liệu theo thứ tự thời gian và lưu vào file ('daxuly.csv')

```
df = df.sort_values('time')
df = df.drop_duplicates(subset='post_id')
df = df.reset_index(drop=True)
df.to_csv('Data/daxuly.csv')
```

```
#output: df.info()
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 1360 entries, 0 to 1359
Data columns (total 26 columns):
 #   Column           Non-Null Count  Dtype  
 ---  -- 
 0   original_request_url    1360 non-null   object 
 1   post_url                 1360 non-null   object 
 ...
 24  page_id                  1360 non-null   int64  
 25  image_ids                1360 non-null   object 
 dtypes: datetime64[ns](1), float64(2), int64(5), object(18)
memory usage: 276.4+ KB
```

Như vậy chỉ còn lại 1360 bài

- Lưu ý : khi crawl user like và user comment ta crawl sau khi xử lý dữ liệu về post ở trên để rút ngắn lại số bài cần crawl phù hợp với bài báo cáo này (so với 1651 bài lúc đầu)

Thêm cột 'has_video','has_image' để xem bài viết đó có ảnh hay video hay không

```
df['image_ids'] = df['image_ids'].apply(ast.literal_eval)
df['image_ids'].apply(pd.Series)
# Thêm cột 'has_video'
df['has_video'] = df['video_id'].apply(lambda x: 0 if np.isnan(x) else 1)

# Thêm cột 'has_image'
df['has_image'] = df['image_ids'].apply(lambda x: 1 if (x and len(x) > 0) else 0)
```

Thêm các cột về lượt reaction

```
import pandas as pd
import numpy as np
import ast

# Chuyển đổi chuỗi thành từ điển
df['reactions'] = df['reactions'].apply(ast.literal_eval)

# Chuyển đổi từ điển thành DataFrame
df_reactions = df['reactions'].apply(pd.Series)

df_reactions = df_reactions.drop(['thich', 'yêu thích', 'thương thương', 'buồn', 'phẫn nộ'], axis=1)
df_reactions=df_reactions.fillna(value=0) #xử lý các số liệu bị null và đặt thành 0
df = pd.concat([df, df_reactions], axis=1)
df.drop(['reactions'], axis=1)
# Tạo cột 'other' bằng cách cộng các cột 'wow', 'care', 'sad', và 'angry'
df['other'] = df[['wow', 'care', 'sad', 'angry']].sum(axis=1)

df = pd.read_csv('Data/chuaxuly.csv') #lưu lại vào file
```

3.Xử lý dữ liệu user like ,comment

Một số dữ liệu bị lặp khi crawl như dữ liệu trong trường "Nơi từng sống" lại nằm trong trường "Name", dữ liệu "Học vấn" lại nằm trong trường "Nơi từng sống", ta xử lý như sau:

```
#tim những cụm từ liên quan đến học vấn trong nơi từng sống để lưu lại vào trường học vấn,và từ trường name vào trường nơi từng sống
def convert_name_to_live_in(df):
    for phrase in ["Học viện", "Đại Học", "Dai hoc", "Đại học", "đại học", "trung cấp", "Trung Cấp", "trung học", "Cao đẳng", "Đại học", "tiến sỹ", "thạc sỹ", "Tiến sỹ", "THPT", "THCS", "Trung học", "Trung hoc", "Trung học"]:
        df.loc[df['Name'].str.contains(phrase, na=False), 'Học vấn'] = df['Nơi từng sống']
    for phrase in ["Quê quán", "Thành phố hiện tại"]:
        df.loc[df['Name'].str.contains(phrase, na=False), 'Nơi từng sống'] = df['Name']
    return df
# Sử dụng hàm
df = convert_name_to_live_in(df)
df['Nơi từng sống'] = df['Nơi từng sống'].apply(lambda x: np.nan if ("Quê quán" not in str(x)) and ("Thành phố hiện tại" not in str(x)) else x)
```

```
# Chỉ giữ lại các cột cần thiết,tá sít phân tích name ở phần dữ liệu khác(dữ liệu gồm comments_pro và likes_pro) đầy đủ hơn
filtered_df = df[['id', 'Nơi từng sống']]

filtered_df.info()
#output :
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 18568 entries, 0 to 18567
Data columns (total 2 columns):
 #   Column           Non-Null Count  Dtype  
 ---  -- 
 0   id               18568 non-null   object 
 1   Nơi từng sống    10149 non-null   object 
 dtypes: object(2)
memory usage: 290.3+ KB
```

Chọn những trường cần thiết

Lưu vào csv

```
# Xóa các dòng trùng lặp dựa trên trường 'id'
filtered_df = filtered_df.drop_duplicates(subset='id')

# Xóa các dòng chứa giá trị NaN trong trường 'id'
filtered_df = filtered_df[filtered_df['id'].notna()]
filtered_df = filtered_df.reset_index(drop=True)
# Lưu DataFrame vào file CSV
filtered_df.to_csv('Data/filtered_data.csv', index=False)
```

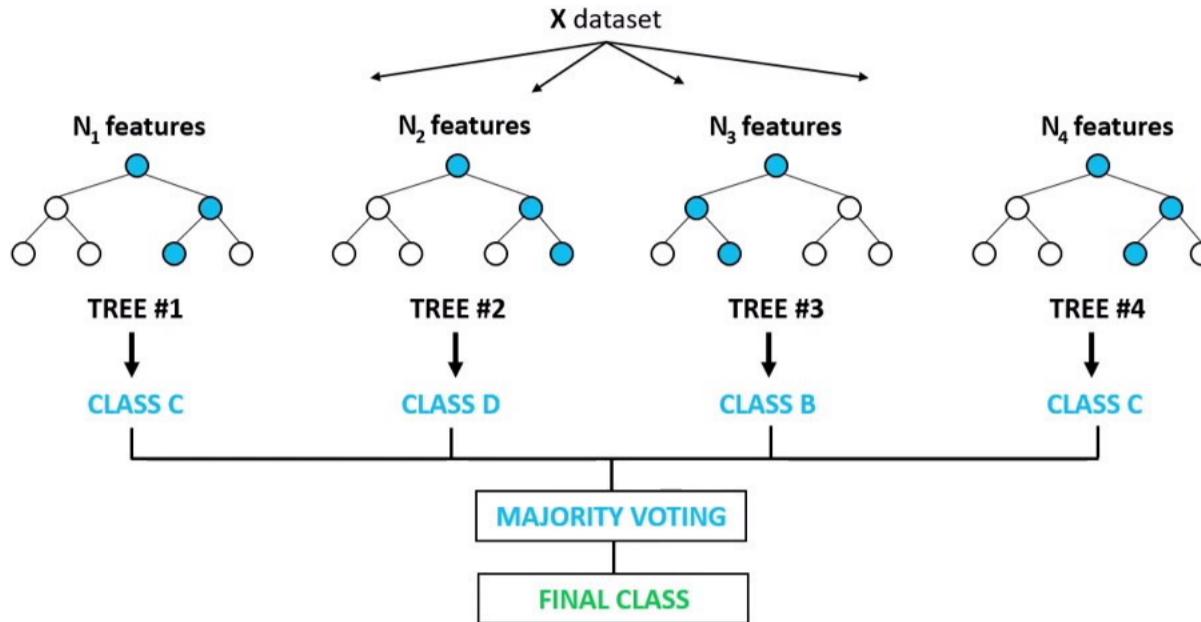
4.Xây dựng các mô hình dự đoán

4.1 Xây dựng mô hình dự đoán bình luận tích cực và tiêu cực

Mô hình được sử dụng là RandomForestClassifier. Random forest là một phương pháp thống kê mô hình hóa bằng máy (machine learning statistic) dùng để phục vụ các mục đích phân loại, tính hồi quy và các nhiệm vụ khác bằng cách xây dựng nhiều cây quyết định (Decision tree).Random Forest cho thấy hiệu quả hơn so với thuật toán phân loại thường được sử dụng vì có khả năng tìm

ra thuộc tính nào quan trọng hơn so với những thuộc tính khác. Trên thực tế, nó còn có thể chỉ ra rằng một số thuộc tính là không có tác dụng trong cây quyết định

Random Forest Classifier



```
import pandas as pd
from sklearn.model_selection import train_test_split
from sklearn.feature_extraction.text import CountVectorizer
from sklearn.ensemble import RandomForestClassifier
from sklearn.metrics import accuracy_score

df=pd.read_csv('Data/dataset.csv')

# Chia dữ liệu thành tập huấn luyện và tập kiểm tra
X_train, X_test, y_train, y_test = train_test_split(df['comment'], df['label'], test_size=0.2, random_state=42)

# Chuyển đổi văn bản thành ma trận đếm từ
vectorizer = CountVectorizer()
X_train_counts = vectorizer.fit_transform(X_train)

# Huấn luyện mô hình RandomForest
clf = RandomForestClassifier(n_estimators=100, random_state=42)
clf.fit(X_train_counts, y_train)

# Chuyển đổi văn bản kiểm tra thành ma trận đếm từ
X_test_counts = vectorizer.transform(X_test)

# Dự đoán nhãn cho tập kiểm tra
y_pred = clf.predict(X_test_counts)

# In ra độ chính xác của mô hình
print("Độ chính xác của mô hình: ", accuracy_score(y_test, y_pred))
#output: Độ chính xác của mô hình: 0.7589001907183726
```

Nguồn dataset : <https://www.kaggle.com/datasets/linhlpv/vietnamese-sentiment-analyst/>

4.2 Xây dựng mô hình dự đoán giới tính

Ta cũng sử dụng mô hình Random Forest để dự đoán trong trường hợp này

```
import pandas as pd
from sklearn.model_selection import train_test_split
from sklearn.ensemble import RandomForestClassifier
from sklearn.feature_extraction.text import CountVectorizer

# Đọc dữ liệu từ file CSV
data = pd.read_csv('Data/UIT-ViNames - Full.csv')

# Chuyển đổi tên thành các đặc trưng bằng cách sử dụng CountVectorizer
vectorizer = CountVectorizer()
X = vectorizer.fit_transform(data['Full_Name'])

# Chia dữ liệu thành tập huấn luyện và tập kiểm tra
X_train, X_test, y_train, y_test = train_test_split(X, data['Gender'], test_size=0.2, random_state=42)

# Xây dựng mô hình Random Forest
model = RandomForestClassifier(n_estimators=100)
model.fit(X_train, y_train)

# Đánh giá độ chính xác của mô hình trên tập kiểm tra
accuracy = model.score(X_test, y_test)

print("Độ chính xác của mô hình là: ", accuracy)
#output : Độ chính xác của mô hình là: 0.9491714764475889
```

Phần 3. Phân tích dữ liệu

• Chuẩn bị phân tích dữ liệu

Giới thiệu về 2 thư viện tôi dùng sau đây để phân tích dữ liệu:

Bài báo cáo này tôi sẽ dùng Matplotlib và Seaborn để phân tích dữ liệu. Đây là hai thư viện Python phổ biến để vẽ đồ thị và trực quan hóa dữ liệu. Matplotlib là một thư viện cơ bản và linh hoạt, cho phép tạo ra nhiều loại đồ thị khác nhau, từ biểu đồ đường đơn giản đến biểu đồ phân tán 3D. Seaborn là một thư viện cao cấp hơn, được xây dựng trên Matplotlib, cung cấp nhiều chức năng và kiểu đồ thị hữu ích cho phân tích thống kê và học máy. Seaborn cũng có giao diện thân thiện hơn và có thể tạo ra các đồ thị đẹp mắt hơn với ít mã hơn. Cả hai thư viện đều có thể làm việc với các khung dữ liệu Pandas và NumPy, hai thư viện khác phổ biến trong Python để xử lý dữ liệu.

Cài đặt các thư viện vẽ biểu đồ và trực quan hóa dữ liệu:

```
%pip install matplotlib  
%pip install seaborn
```

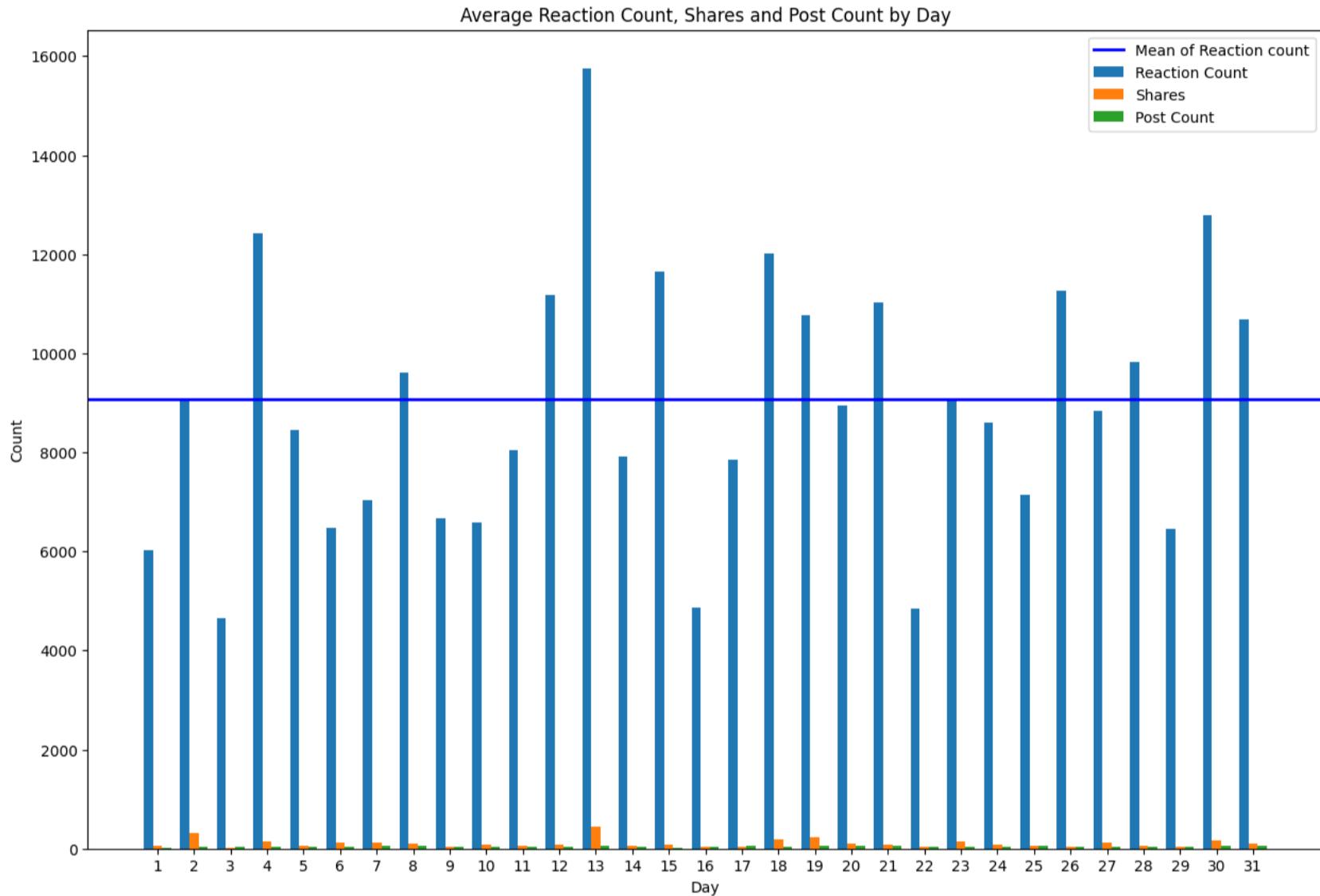
1. Phân tích về lượng tương tác của các bài viết

1.1. Trung bình số lượt like, share, comment

Trung bình số lượng bài đăng theo ngày: 43.87096774193548
Trung bình số lượng phản ứng trong 31 ngày: 8919.814822789536
Trung bình số lượng chia sẻ theo ngày: 106.81049297417164

- Số lượng bài đăng trung bình mỗi ngày là **43.87**: cho thấy page đang duy trì một lượng nội dung đều đặn để thu hút người dùng.
- Số lượng phản ứng trung bình mỗi bài đăng là **8919.81**: Một con số khá là tích cực với 1 fanpage tin tức hàng đầu trên mạng xã hội hiện nay
- Số lượng chia sẻ trung bình là **106.81**. Điều này cho thấy các bài viết đang được lan truyền rộng rãi trong cộng đồng.

1.2 Biểu đồ thể hiện lượng tương tác của page trong tháng



Từ biểu đồ cho thấy trang Theanh28 hoạt động khá tích cực và ổn định, trung bình các bài viết đều có lượt tương tác cao

💡 Các ngày 4, 13, 21, 30 có số lượt tương tác rất cao ta sẽ phân tích xem trong ngày đó phân tích đăng những gì

Post_Text	Reaction_Count
Nhin từ xa cứ ngỡ là 1 căn biệt thự 😊 Cho đi rồi sẽ được nhận lại sự tử tế <3 Sơn Tùng M-TP biểu diễn "Em của ngày hôm qua" ... Ấm lòng, người tốt luôn ở khắp mọi nơi 😍	137013 369489 196931 178616

Post_Text:
Nhìn từ xa cứ ngỡ là 1 căn biệt thự 😊
Cho đi rồi sẽ được nhận lại sự tử tế <3
Sơn Tùng M-TP biểu diễn "Em của ngày hôm qua" ...
Ấm lòng, người tốt luôn ở khắp mọi nơi 😍

Reaction_Count:
137013
369489
196931
178616

Left Post Content:
VIDEO QUYỀN BOUTIQUE
Sự tử tế từ bên ngoài
...
CHO CÔ MUỐN CHỖ ĐỂ BÁN HÀNG TRƯỚC QUÁN, CHỦ TIỆM CẢM BÓNG KHI LUÔN BUỘC CÔ LAU DỌN GIÚP TRƯỚC KHI TỐI GIỜ MỞ CỬA
THEANH28 NEWS

Right Post Content:
NGUỒN: MNBNH3
THAY 2 MẸ CON VÀO QUÁN ĂN CHỈ DÁM GỌI 1 BÁT BÚN 10K, NGƯỜI ĐÀN ÔNG LIÊN GIÚP BƠ KHÍEN AI CÙNG CẨM THẤY ẤM LÒNG
THEANH28 NEWS



- Ta thấy 3/4 bài viết đều là các bài viết tích cực về những việc làm tử tế, những người Việt có tấm lòng nhân hậu luôn làm điều tốt đẹp. Điều đó nhắc nhở mọi người cần làm việc tốt trong cuộc sống hàng ngày. Vì vậy nó có thể thu hút nhiều tương tác vì mọi người muốn chia sẻ cảm xúc tích cực này với người khác.
- Trong các bài viết có nhiều lượt tương tác nhất xuất hiện 1 ca sĩ quen thuộc đó là Sơn Tùng MTP - anh là ca sĩ cũng như cá nhân đầu tiên đạt được 10 triệu follow trên trang facebook, hiện tại con số đã lên đến 14 triệu gần bằng 1/7 dân số Việt Nam hiện nay. Anh là ca sĩ có tầm ảnh hưởng lớn nhất trong 10 năm trở lại đây tại Việt Nam bức ảnh trên cũng đánh dấu 10 năm anh tham gia chương trình VietNam Idol - cánh cửa đầu tiên của anh khi bước chân vào showbiz

SƠN TÙNG M-TP

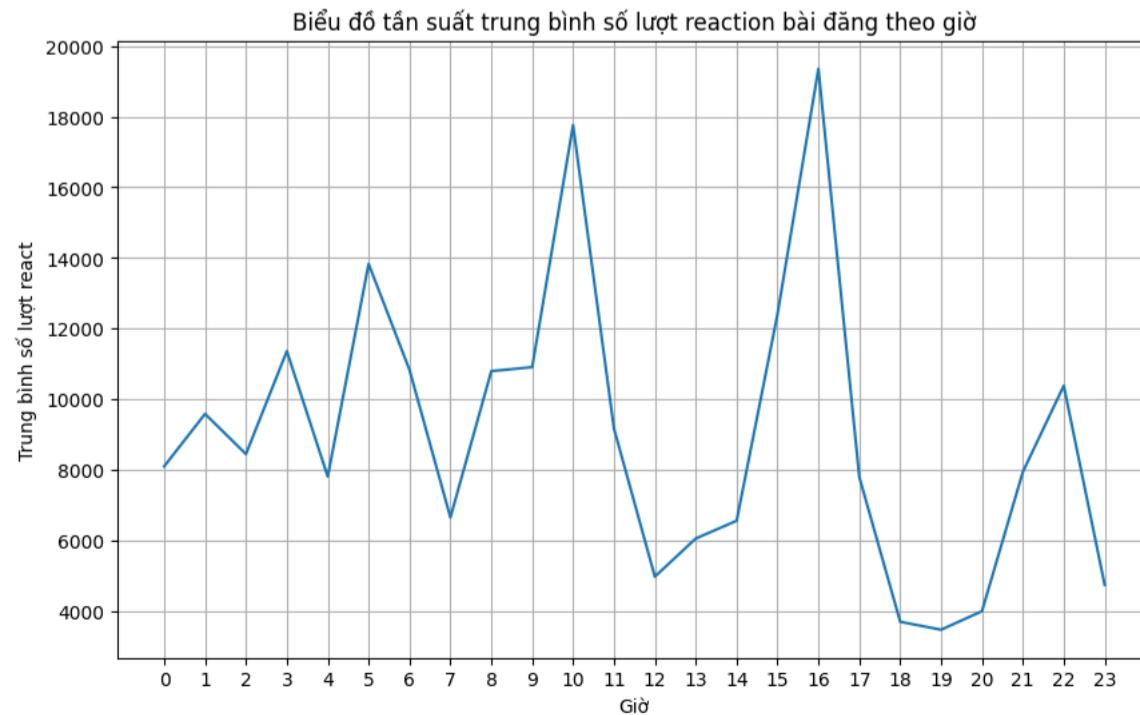
SONG HÒA

M-TP

14 triệu người theo dõi • 5 đang theo dõi

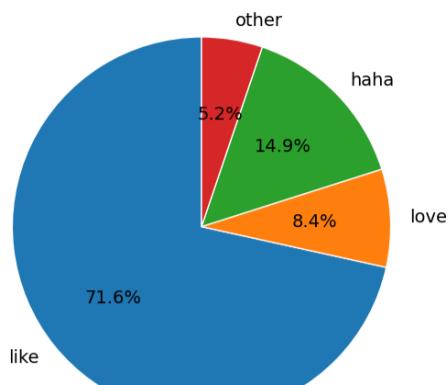
Xem ngay Nhắn tin Theo dõi

1.3 Biểu đồ trung bình số lượt reaction bài theo giờ trong ngày

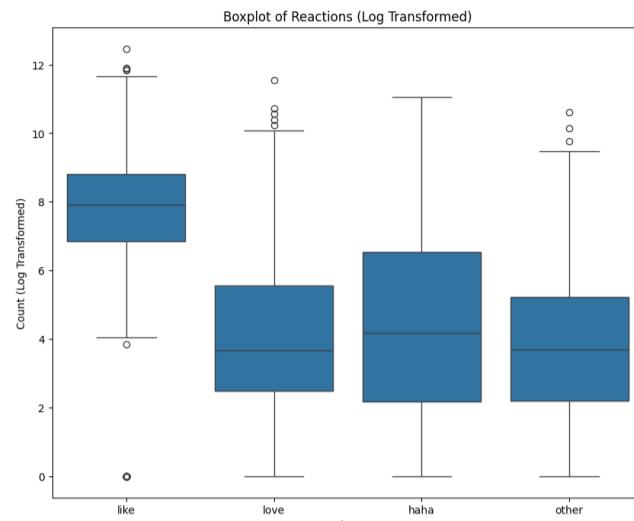


Ta nhận thấy các bài đăng có số lượt tương tác nhiều nhất thường là vào lúc giữa trưa, chiều muộn và buổi tối. Đây là những khung giờ ngoài giờ làm việc và số lượng tương tác sẽ cao hơn những khung giờ khác

1.4 Biểu đồ tương quan giữa tỉ lệ các loại emote



Biểu đồ tròn



Biểu đồ hòn được phỏm theo từng loại phản ứng

Vì dữ liệu có nhiều bài có số lượt reaction cao đột biến nên ta tính logarit của chúng

Chú thích : like(thích) love(tâm) **haha** other gồm care(ther/gher/điều), sad(buồn), angry(tức giận)

Ta có thể phân xem như sau:

- Page có mức độ tương tác cao và tích cực từ người dùng. Lượt like chiếm tỷ lệ lớn nhất, cho thấy nhiều người quan tâm và đánh giá cao nội dung của họ. Lượt haha cũng chiếm tỷ lệ cao, cho thấy page có nhiều nội dung hài hước, giải trí cho người xem. Đây là một yếu tố thu hút và giữ chân người dùng.
 - Lượt other chiếm tỷ lệ thấp nhất, cho thấy page ít có nội dung gây tranh cãi, phản ứng tiêu cực hoặc không phù hợp với người dùng.

3. Phân tích nôi dung bài viết

2.1 Các từ/baoban đượcdùngnhiều nhấtt



Các từ được dùng nhiều nhất

Các từ xuất hiện nhiều nhất trên một page tin tức sẽ phản ánh nội dung chính, độc giả mục tiêu cũng như phong cách viết của page đó.

- Các từ xuất hiện nhiều nhất thường là những từ thường dùng trong văn bản hàng ngày và đều là những từ dùng để nối. Ví dụ, các từ như “đã”, “và”, “với”, “của” đều dùng để nối các câu lại với nhau thành 1 văn bản hoàn chỉnh nên sẽ xuất hiện nhiều trong ngữ cảnh hàng ngày



Các hashtag được dùng nhiều nhất

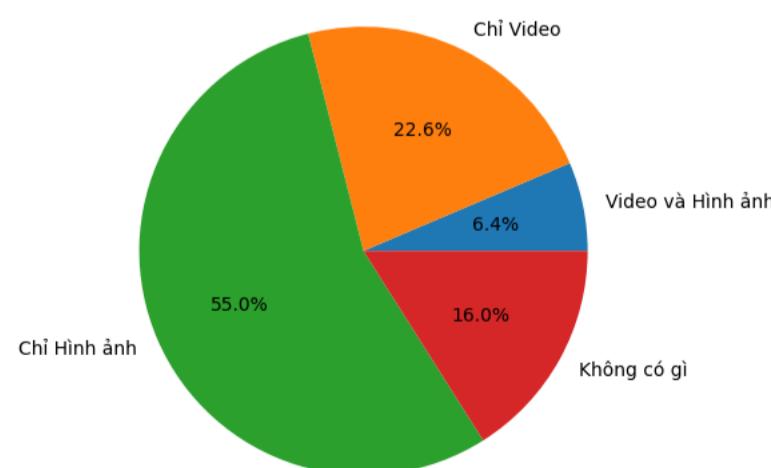
Nổi bật nhất đó là các hashtag #MissGrandInternational2023, #MGI2023, #MissGrandInternational

Với sự xuất hiện của cuộc thi Hoa hậu Hòa bình Quốc Tế 2023(MissGrandInternational2023) diễn ra vào tháng 10/2023 thì sự xuất hiện phần lớn các hashtag liên quan đến cuộc thi này là điều dễ hiểu



Hình ảnh từ cuộc thi MGI 2023

2.2 Phân tích ảnh và video



Biểu đồ tròn thể hiện sự xuất hiện của hình ảnh và video trong bài viết

Ta nhận thấy phần lớn các bài viết chứa hình ảnh hoặc video và chỉ có **16%** những bài viết chỉ gồm văn bản. Điều đó cho thấy hình ảnh và video là công cụ diễn đạt nội dung phổ biến nhất trong bài viết, bởi nó có thể thu hút sự chú ý của người theo dõi và làm nổi bật nội dung của bài viết.cung cấp thông tin chi tiết và sinh động hơn

3. Phân tích user comment, user like

3.0 Nhân vật bình luận nhiều nhất

Người có số lượng bình luận trung bình cao nhất là Theanh28 Entertainment (100069153349307) với trung bình 723 bình luận.

警示教育 ngay nhiên khi người bình luận nhiều nhất lại chính là page Theanh 28

👉 Phân tích page đã bình luận về chủ đề gì :

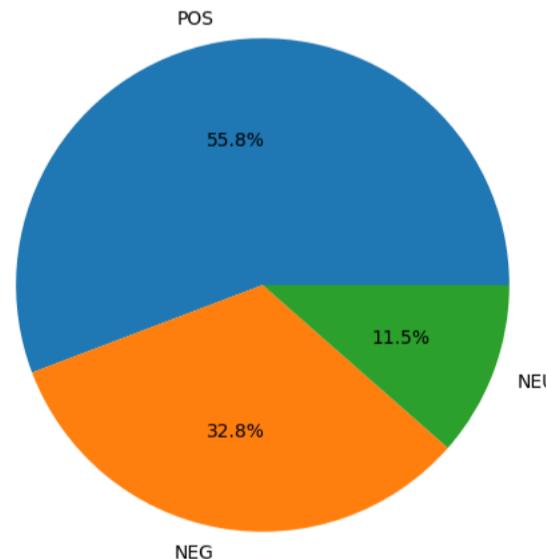
profileName	text	postTitle
0 Theanh28 Entertainment	Nhin ảnh nạn nhân mà đau lòng... Chi tiết vụ việc tại Báo Nhân Dân nhé https://nhandan.vn/nan-nhan-thu-5-tu-vong-trong-vu-tai-nan-giao-thong-tren-quoc-lo-20-post775396.html	Được biết, con gái nạn nhân thứ 5 tử vong mới 18 tuổi, đang được điều trị tại Bệnh viện đa khoa Đồng Nai trong tình trạng nguy kịch. Quá thương tâm! Chi tiết vụ việc dưới phần bình luận nhé ↗
1 Theanh28 Entertainment	Trước tình huống cấp thiết, Phòng Cảnh sát Phòng cháy chữa cháy và Cứu nạn cứu hộ đã huy động một xe thang, một xe cứu nạn cùng 15 cán bộ, chiến sĩ đến hiện trường.\n\nCảnh sát nhận định cửa bị kh...	DÙ TRỜI MƯA VÀ PHẢI THỰC HIỆN NHIỆM VỤ Ở ĐỘ CAO 60M, CẢNH SÁT ĐÃ TIẾP CẬN HIỆN TRƯỜNG CHỈ TRONG VÀI PHÚT VÀ GIẢI CỨU THÀNH CÔNG CHÂU BÊN Nắng Sáng 1/10, thông tin từ Phòng Cảnh sát phòng cháy chữa ch...
2 Theanh28 Entertainment	Ths.Bs Mai Thị Anh Thư (Bệnh viện Mắt Hà Nội 2) cho biết, bệnh nhân bị cận loạn thị nặng từ bé do ảnh hưởng bởi gene di truyền khiến nhãn cầu có kích thước lớn hơn và lồi mắt hơn hẳn so với mắt ng...	NGÃ XE ĐẬP ĐỊEN, NGƯỜI ĐÀN ÔNG BỊ RƠI THỦY TINH THỂ, CỐ THẾ MỦ VĨNH VIỄN\nTheo đó, anh N.D.T (31 tuổi, Gia Lâm, Hà Nội) ngã đập mặt xuống lòng đường, mất bên phải bị rơm thủy tinh thể buộc ...
3 Theanh28 Entertainment	Trước đó, khoảng 2h40 cùng ngày, xe ô tô khách mang biển kiểm soát 50F-004.83 do H.V.T (SN 1986, quê tỉnh Thừa Thiên Huế), điều khiển chở theo 34 hành khách, lưu thông trên quốc lộ 20 hướng từ Dầu...	DÙ ĐÃ BỊ TƯỚC BẰNG LÁI NHƯNG TÀI XE VẪN DIỄU KHIỂN XE GÂY TAI NAN\nNgày 1/10, Công an tỉnh Đồng Nai thông tin, sẽ mở rộng điều tra rõ trách nhiệm chủ xe Thành Buối và những người khác liên qua...
4 Theanh28 Entertainment	CHÙM ẢNH TUYỆT ĐẸP CARNAVAL THU HÀ NỘI TẠI ĐÂY:\n https://nhandan.vn/anh-ru-ro-sac-mau-carnaval-thu-ha-noi-tren-pho-di-bo-ho-hoan-kiem-post775421.html	Sáng 1/10, Carnaval Thu Hà Nội nằm trong khuôn khổ Festival Thu Hà Nội đã diễn ra vô cùng sôi động tại phố đi bộ hồ Hoàn Kiếm.\nNhững tiết mục biểu diễn được dàn dựng công phu với quy mô hơn 1.5...
...
718 Theanh28 Entertainment	Em Dứt ngủ ngoan nhé ❤️	Tạm biệt Dứt, thời gian qua em đã mang lại niềm vui cho rất nhiều người ❤️
719 Theanh28 Entertainment	Ăn xong vẫn đòi 😊	BÚA COM LÈO TÈO 32K CỦA HỌC SINH TRƯỜNG BẢN TRŨ\nMới đây phụ huynh Trường THCS Yên Nghĩa đã bắt ngờ kiểm tra bếp ăn và vô cùng bức xúc khi chứng kiến bữa ăn của các con chỉ "lèo tèo" vài món.\n...
720 Theanh28 Entertainment	Giám đốc Sở GD-ĐT TP.HCM khẳng định "không có khái niệm quỹ lớp quỹ trường".	SỞ GD-ĐT TP.HCM SẼ THANH TRA TỪ NGÀY 16/10 - 16/11\nSở GD-ĐT TP.HCM sẽ thanh tra, kiểm tra, giám sát tình hình công tác quản lý thu, chi đầu năm học tại các cơ sở giáo dục; Kiểm tra công tác ...
721 Theanh28 Entertainment	Cho ai muốn đọc chi tiết câu chuyện thì vào đây nhé https://www.facebook.com/groups/cafeduongpho.vn/?sorting_setting=CHRONOLOGICAL	MÃI MÃI LÀ CHÍ TỬ THỨ 7 ĐẾN HẾT CHỦ NHẬT\nMới đây, 1 chị vợ đã chia sẻ câu chuyện mà chẳng có 1 người phụ nữ nào muốn gặp phải khi lấy chồng. Được biết, con bị ốm phải nhập viện nên chị 1 mình v...
722 Theanh28 Entertainment	Link đặt hàng online sản phẩm:\nWebsite: https://gaudau.yadea.com.vn/ \nShopee: https://shopee.vn/yadea2023 \nLazada: https://bit.ly/3QCFLZH	MÙA ĐÔNG, NGƯỜI YÊU CÓ THỂ KHÔNG CÓ NHỮNG NHẤT ĐỊNH PHẢI CÓ MỘT CHÚ "GẦU NHỎ YADEA" 🐾\nThu qua đông tới, người yêu có thể không có nhưng nhất định phải sở hữu một chú "gấu nhỏ YADEA" nhé. Bạn b...

723 rows x 3 columns

Ta nhận thấy những bình luận của page điều liên quan đến bài viết, một phần nhằm bổ sung thêm những ý chính cho bài đăng một phần là đính kèm link tránh kiểm duyệt facebook cũng như đánh vào thói quen người dùng hay "soi" bình luận giúp tăng tương tác. Một câu nói rất quen thuộc trên facebook : "chi tiết vụ việc dưới phần bình luận"

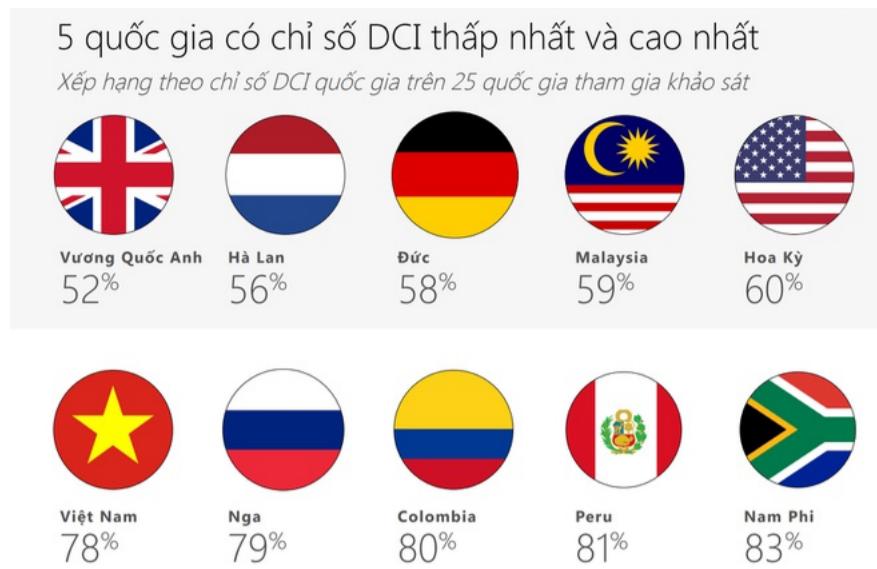
3.2 Phân tích lượng bình luận tích cực và tiêu cực

Tương quan số lượng bình luận tích cực và tiêu cực



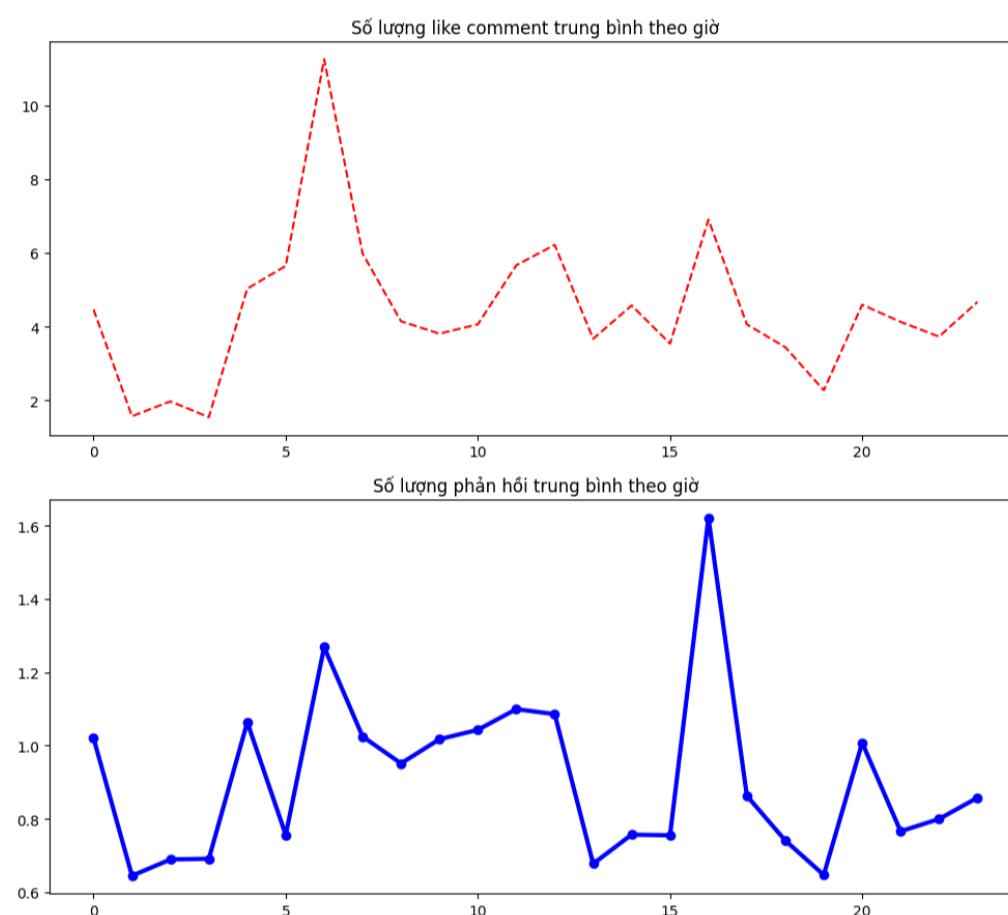
- Chú thích : **POS**(Tích cực), **NEU**(Bình thường/Trung lập), **NEG**(Tiêu cực)

Từ biểu đồ ta thấy lượng bình luận tích cực chiếm đa số nhưng chúng ta cũng không nên vui mừng vì điều đó, nhìn sang lượng bình luận tiêu cực chiếm đến gần 33% nghĩa là cứ 3 bình luận lại có 1 bình luận tiêu cực một con số đáng ngại. Điều đó cho thấy Việt Nam là một quốc gia kém văn minh trên Internet, thật vậy theo khảo sát Việt Nam thuộc top 5 quốc gia kém văn minh mạng trên thế giới



Theo khảo sát mới được công bố của Microsoft, Việt Nam nằm trong top 5 quốc gia có chỉ số mức độ văn minh thấp nhất trên không gian mạng (DCI). Kết quả này được Microsoft công bố nhân ngày quốc tế An toàn Internet. Hiện Việt Nam đứng thứ 5 sau Nga, Colombia, Peru và Nam Phi.

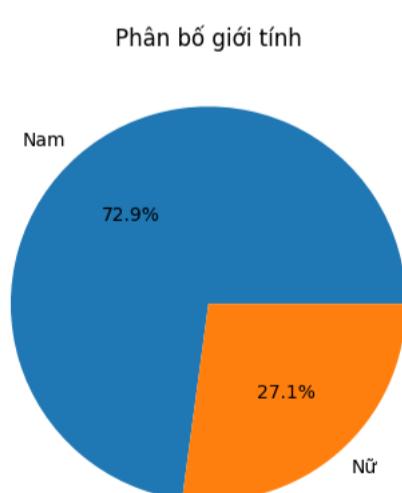
3.1 Phân tích khả năng tương tác của người dùng



Biểu đồ thể hiện khả năng tương tác của người dùng qua số lượng like comment và số lượng phản hồi comment

- Hai biểu đồ có dáng điệu tương tự nhau cho thấy những bình luận được nhiều người quan tâm đều có số lượng lớn react và reply đi kèm, ta thấy khoảng thời gian khi người dùng comment có nhiều lượt tương tác nhất là khoảng 6-8h sáng hoặc 16-18h chiều tối đây là những khoảng thời gian ngoài giờ làm việc, vậy nên nếu ta muốn bình luận bài viết nào đó nên chọn 2 khung giờ trên để có tỉ lệ người dùng phản hồi lại chúng ta cao nhất(hay là đỡ bị "quê")

3.3 Phân tích sự tương quan giữa các giới tính khi sử dụng facebook(dựa trên dữ liệu người like và comment)



Biểu đồ tròn thể hiện sự tương tác khác nhau giữa 2 giới tính nam và nữ

Thật bất ngờ khi đa số các bài viết số lượng người có giới tính Nam lại tham gia tương tác nhiều hơn chiếm gần 3/4, trái ngược với quan điểm phụ nữ hay sống ảo. Phải chăng tỉ lệ chênh lệch dân số giữa nam và nữ ở Việt Nam lại ảnh hưởng đến điều này?

Câu trả lời là không đáng kể, có thể nữ giới thích đăng bài hơn nhưng lại ngại tham gia tương tác trong khi đó nam giới lại tham gia tương tác nhiều hơn, điều đó cho thấy nam giới thích đánh giá hơn và phụ nữ thích được nhận sự đánh giá hơn khi họ tham gia Facebook 😊

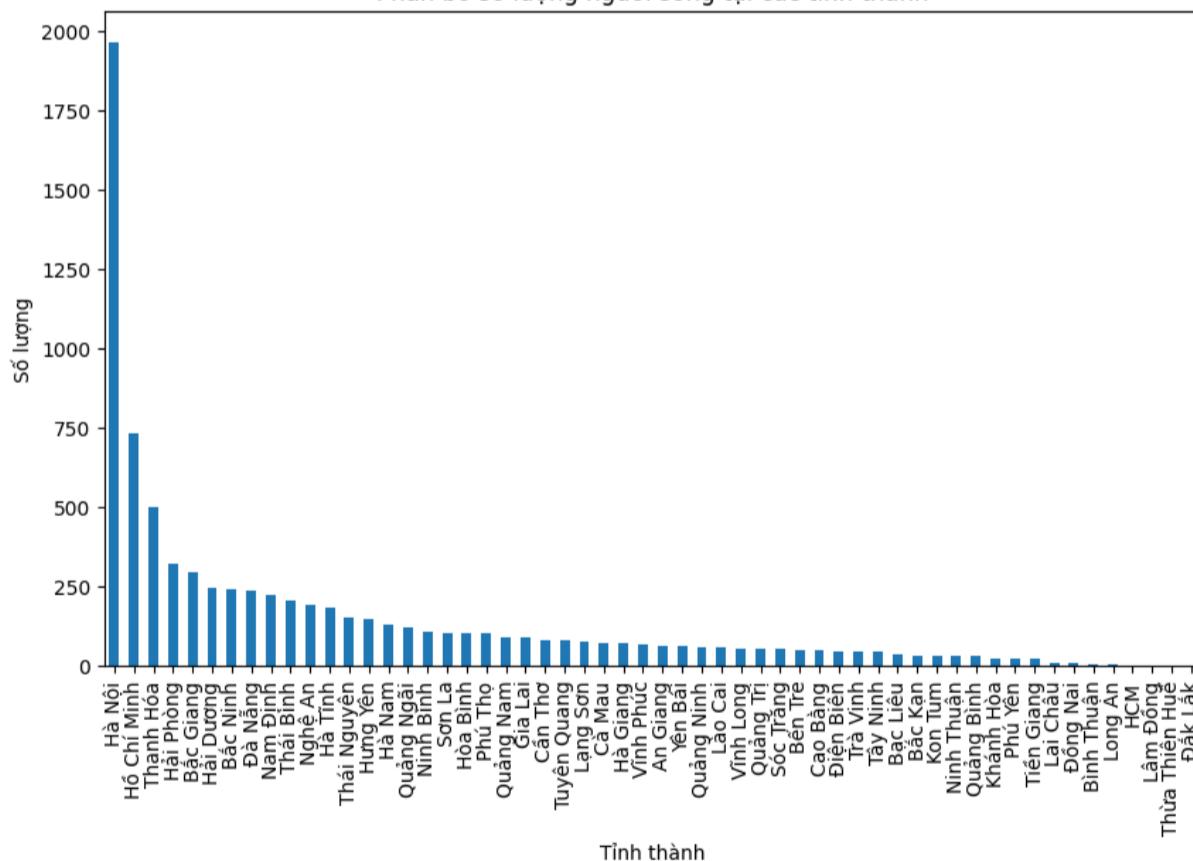
3.4 Những cái tên xuất hiện nhiều trên facebook



Ta thấy có những điểm lưu ý là xuất hiện một số họ phổ biến ở Việt Nam như họ Nguyễn|Nguyen,Lê,Trần,Bùi,Phạm
Những cái tên xuất hiện nhiều nhất là Hoàng, Minh, Vũ, Linh,...(những cái tên con trai hiển thị rõ hơn chứng tỏ nam giới chiếm đa số)

3.4 Phân tích sự tương quan về số lượng người ở những tỉnh thành tham gia facebook

Phân bố số lượng người sống tại các tỉnh thành



Biểu đồ thể hiện nơi sinh sống của những người tham gia tương tác vào trang Theanh28

Nhận thấy thủ đô Hà Nội chiếm một tỉ lệ vô cùng lớn gần bằng tổng các tỉnh thành khác cộng lại , điều đó cho thấy mật độ dân số ở Hà Nội khá đông cũng như trình độ khá phát triển, tiếp theo đó là thành phố Hồ Chí Minh, đây đều là 2 thành phố đông dân và có trình độ phát triển nhất nước ta.

Kết luận

Qua bài báo cáo ta đã đi qua những vấn đề cả về việc xử lí cũng như phân tích dữ liệu. Ta cũng nhận ra những thông tin thú vị từ những số liệu mà ta thu thập được từ page qua đó như nền đăng bài giờ nào để đạt nhiều tương tác nhất, tỉ lệ giữa con trai và con gái khi sử dụng facebook để tương tác hay những từ khóa phổ biến ẩn đăng sau mỗi bài viết. Tuy là một bài báo cáo nhỏ nhưng cũng có một phần lợi ích dành cho những nhà sáng tạo nội dung muốn tìm hiểu thêm thị phần để phát triển nội dung của mình.

Song khi làm bài báo cáo này tôi cũng không gặp khó khăn như bị gắn spam flag liên tục và đặc biệt thời gian chuẩn bị không nhiều nên việc lựa chọn các mô hình dự đoán còn đơn giản hay chỉ crawl dữ liệu trong vòng 1 tháng

Dù vậy nhưng bài báo cáo này cũng đã đáp ứng đủ những thông tin cần thiết, một phần nào cũng sẽ có tác dụng cho chúng ta cho người đọc.

Thank you for reading !!